4Б

УДК 004.93’11

О.В. Бармак, Ю.В. Крак, Ю.Г. Кривонос
Інститут кібернетики ім. В.М. Глушкова НАН України, м. Київ, Україна
yuri.krak@gmail.com

Про один підхід до ідентифікації особи за контуром профілю носа

У статті розглядається підхід для ідентифікації особи за контуром профілю носа. Для знаходження простору ознак пропонується використовувати апроксимацію за допомогою *B*-сплайнів. Алгоритмічна реалізація підходу використовує моделювання структури бази даних у вигляді *n*- вимірного куба.

Вступ

Біометрична ідентифікація побудована на порівнянні деяких параметрів (як фізіологічних, так і психологічних), що властиві людині та занесені до бази даних. У випадку фізіології для аналізу та порівняння використовуються такі постійні та незмінні параметри людини, як фізичні характеристики: риси і форма обличчя, особливості очей, рук – долоні та пальці (папілярний узор, топографія, форма), венозний рисунок, теплова картина тощо.

Останнім часом у дослідників викликає інтерес і вважається перспективною ідентифікація людини за формою носа [1]. Одним із варіантів застосування такого способу встановлення особи є виявлення злочинців, знятих на камери прихованого спостереження.

Алгоритми локалізації облич на зображенні та виділення контурів на обличчі досить розвинені. **Метою даної роботи** є задача ідентифікації особи за контуром профілю носа, за умови, що такий контур вже отримано. Матимемо на увазі, що процедури отримання контуру (навіть за умови його віднормованості) не гарантують повний збіг в однієї і тієї ж особи. Це випливає із особливостей як зображення, так і цих процедур. Тому розв’язок задачі ідентифікації має це враховувати і вирішувати.

Формалізація підходу

Алгоритми обробки зображень дозволяють отримати контур носа як множину точок:

$$Z = \left\{ z_i = (x_i, y_i), i = 0 \cdots \widetilde{M} - 1, x_i, y_i \in R \right\}. \tag{1}$$

Подальша робота з такою множиною точок можлива за умови проведення відповідного нормуючого перетворення:

$$Z \to D, \tag{2}$$

де $D = \left\{ d_i = (x_i, y_i), i = 0 \cdots \widetilde{M} - 1, x_i, y_i \in [0,1] \right\}$.

Пропонується (2) отримати за допомогою:

$$d_i = z_i A, \tag{3}$$

де A – матриця (3×3) перетворення, яка визначається як добуток матриць переносу, повороту та масштабування; $d_i = (x_i, y_i, 1)$, $z_i = (x_i, y_i, 1)$ – точки в однорідній системі координат. Отримаємо матрицю A в аналітичному вигляді. Задача полягає у визначенні матриць: переносу (T) – для переносу кривої-контуру в початок системи координат,

повороту (R) – для суміщення початкової та кінцевої точок кривої з оссю X та масштабування (S) – для розподілу точок кривої між нулем та одиницею:

$$T=\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ -x_0 & -y_0 & 1\end{pmatrix},\ R=\begin{pmatrix}\cos(\alpha) & -\sin(\alpha) & 0\\ \sin(\alpha) & \cos(\alpha) & 0\\ 0 & 0 & 1\end{pmatrix},\ S=\begin{pmatrix}s & 0 & 0\\ 0 & s & 0\\ 0 & 0 & 1\end{pmatrix}, \qquad (4)$$

де $\cos\alpha=\left(x_{\widetilde{M}-1}-x_0\right)/Q$; $\sin\alpha=\left(y_{\widetilde{M}-1}-y_0\right)/Q$; $Q=\sqrt{\left(x_0-x_{\widetilde{M}-1}\right)^2+\left(y_0-y_{\widetilde{M}-1}\right)^2}$ – відстань між крайніми точками кривої Z; $s=1/Q_x$; $Q_x=x_{\widetilde{M}-1}-x_0$.

Матриця перетворення A матиме вигляд:

$$A=T\cdot R\cdot S=\begin{pmatrix}s\cos(\alpha) & -s\sin(\alpha) & 0\\ s\sin(\alpha) & s\cos(\alpha) & 0\\ -s\left(y_0\sin(\alpha)+x_0\cos(\alpha)\right) & -s\left(x_0\sin(\alpha)-y_0\cos(\alpha)\right) & 1\end{pmatrix}=$$

$$=\begin{pmatrix}1/Q & -Q_y/(Q_x\cdot Q) & 0\\ Q_y/(Q_x\cdot Q) & 1/Q & 0\\ -(x_0\cdot Q_x+y_0\cdot Q_y)/(Q_x\cdot Q) & (x_0\cdot Q_x-y_0\cdot Q_x)/(Q_x\cdot Q) & 1\end{pmatrix}, \qquad (5)$$

де $Q_y=y_{\widetilde{M}-1}-y_0$.

Процедури визначення точкових кривих, що відтворюють контури носів різних людей, продукують різну кількість точок ($\widetilde{M}$). Подальші перетворення можливі за умови рівності кількості точок (M) у кривих-контурах. Досягнути рівності кількості точок можливо, скориставшись кусково-лінійною інтерполяцією:

$$\begin{aligned}&x_i=x_{i+1}+\left(x_0-x_{M-1}\right)/M-1\\ &y_i=\widetilde{y}_j\left(\widetilde{x}_{j+1}-x_i\right)+\widetilde{y}_{j+1}\left(x_i-\widetilde{x}_j\right)/\left(\widetilde{x}_{j+1}-\widetilde{x}_j\right),\ j:\widetilde{x}_j<x_i<\widetilde{x}_{j+1}\end{aligned}, \qquad (6)$$

де $x_0=\widetilde{x}_0$, $y_0=\widetilde{y}_0$, $i=1,\cdots,M-1$, $j\in\left\{0,\cdots,\widetilde{M}-1\right\}$; $\widetilde{x},\widetilde{y}$ – вхідна (неінтерпольована) множина.

Поставлена мета полягає у знаходженні відображення множини точок (D) у множину ознак (P):

$$D\to P, \qquad (7)$$

де $P=\left\{p_i=\left(x_i,y_i\right),i=1\cdots N\right\}$, $N<<M$, елементи якої єдиним чином визначають контур носа. Із фізіогноміки, живопису [2] та криміналістичної експертизи [3] відомо, що профіль носа має декілька зон-особливостей, комбінація яких і дає відомі контури носа (грецький, римський, орлиний тощо). До таких зон відносяться основа носа, кінчик носа, спинка носа та ще можна віднести перенісся. Перенісся враховується для більш повної типізації носів. Топологія кривої, яка відтворює контур носа, має яскраво виражені чотири зони особливостей. Звідки, для перетворення (7) – N = 4.

Перетворення (7) пропонується знайти, використовуючи B-сплайн-апроксимацію. Задача B-сплайн-апроксимації є задачею підгонки B-сплайн-кривої з K-опорними точками $p=\left(p_0,\ldots,p_{K-1}\right)^T$ до точкової кривої $d=\left(d_0,\ldots,d_{M-1}\right)^T$, де M>K (зазвичай M>>K) для значень параметра $u_0,\ldots,u_{M-1}$. Така задача апроксимації приводить до перевизначеної системи лінійних рівнянь:

$$N(u)\cdot p=d, \qquad (8)$$

де $N_i(u)$ – B-сплайн-базисна функція.

Підхід, вибраний для формування *B*-сплайнів, полягає у тому, щоб визначати сплайн в термінах множин базисних функцій, кожна з яких відмінна від нуля тільки на інтервалі в декілька вузлів (значень параметра *u*). Кожна функція $N_i(u)$ є поліномом заданого степеня на інтервалі в декілька вузлів та дорівнює нулю за межами інтервалу. Множина функцій $\{N_i(u)\}$ утворює базис для заданої послідовності параметрів *u* і заданої степені полінома.

Існує багато способів визначення базисних функцій ($N_i(u)$), але особливе місце належить рекурсивним функціям Кокса-де Бура [4]:

$$N_{k,0} = \begin{cases} 1, \text{ якщо } u_k \le u \le u_{k+1} \\ 0 - \text{у іншому випадку} \end{cases}, \qquad N_{k,d} = \frac{u - u_k}{u_{k+d} - u_k} N_{k,d-1}(u) + \frac{u_{k+d} - u}{u_{k+d+1} - u_{k+1}} N_{k+1,d-1}(u) \tag{9}$$

де $u_0, u_1, \cdots, u_n$ – послідовність вузлів, така що:

$$u_{\min} = u_0 \le u_1 \le \cdots \le u_n = u_{\max}.$$

Кожна функція із першої множини, $N_{k,0}$, постійна на одному інтервалі та дорівнює нулю за його межами; кожна функція із другої множини, $N_{k,1}$, лінійна на двох інтервалах та дорівнює нулю за їх межами; кожна функція із третьої множини, $N_{k,2}$, має вигляд квадратичної кривої на трьох інтервалах та дорівнює нулю за їх межами. В загальному випадку функція їх множини $N_{k,d}$ має відмінне від нуля значення на $d+1$ інтервалах між u_k і u_{k+d+1} і має вигляд полінома степені *d* на кожному із цих інтервалів, причому у вузлах забезпечується параметрична неперервність класа C^{d-1}. Сплайн-крива, утворена як зважена сума таких базисних функцій, буде лежати всередині опуклої багатокутної оболонки, так як для базисних функцій виконується умова: $\sum_{i=0}^{M-1} N_{i,d}(u) = 1,\ 0 \le N_{i,d}(u) \le 1$. Але від того, що кожна функція $N_{k,d}$ відмінна від нуля тільки на $d+1$ інтервалах, то кожна опорна точка впливає тільки на ту дільницю сумарної кривої, яка лежить всередині оболонки, утвореної $d+1$ опорними точками.

Множина базисних функцій *B*-сплайна визначається заданим степенем сплайна та вектором вузлів. Не накладено ніяких обмежень на значення у вузлах, крім як $u_k \le u_{k+1}$. Якщо вважати, що результат ділення на ноль при обчисленні по рекурсивним формулам (9) дорівнює 1, то можна мати кратні вузли (ті, що повторюються). При повторенні вузлів (кратності вузлів) проявляється ефект наближення сформованого *B*-сплайна до опорної точки, яка асоціюється з цими вузлами. Якщо кратність вузла буде дорівнювати $d+1$, то *B*-сплайн степені *d* буде проходити через відповідну опорну точку.

Формування кубічного сплайна проводиться з використанням наступного масиву вузлів: $\{0, 0, 0, 0..., 1, 1, 1, 1\}$. Наслідком цього є належність першої та останньої опорної точки самій кривій. Тому для отримання потрібних для аналізу *n* точок потрібно взяти *n+2* точки.

Базисні сплайни заданої степені є лінійно незалежними функціями та утворюють базис у функціональних просторах, що можна використовувати для відображення з їх допомогою інших функцій цих же просторів. Виходячи з цього, для отримання набли-

женого розв'язку перевизначеної системи лінійних рівнянь (8) можна скористатися методом найменших квадратів [5], [6]:

$$N^T N \cdot p = N^T \cdot d\,, \tag{10}$$

звідки

$$p = \left(N^T N\right)^{-1} \cdot N^T d\,. \tag{11}$$

Задача відтворення за визначеними за допомогою (11) опорними точками вхідної кривої для профілю носа потребує від 9 до 16 точок. Це пов'язано з тим, що, крім означених вище чотирьох зон-особливостей, у профілів носів є ще ряд незначних (з точки зору особливостей) кривизн. Пропонується визначати чотири опорні точки. З точки зору апроксимації, точного відтворення вхідного профілю не отримаємо, але у цих чотирьох точках будуть сконцентровані особливості саме в потрібних нам чотирьох зонах.

Застосування перетворення (7) для множини контурів носів, які треба ідентифікувати, приводить до отримання чотирьох множин параметрів для зон особливостей профілю носа:

$$P^j = \left\{\left(y_i^j, Ind\right)\right\}, \tag{12}$$

де $j = 1,...,4$ – для чотирьох зон; $i = 1,...,L$, де L – кількість профілів носів різних людей; Ind – номер, який однозначно ідентифікує людину.

При формуванні множин (12) використовуються тільки *y*-координати точок-ознак. Це випливає з того, що перед перетворенням (7) множина точок (контур носа) піддалася кусково-лінійній інтерполяції (6). Як наслідок цього, *x*-координати точок-ознак різних контурів будуть однакові, тому їх можна виключити з подальшого розгляду.

Тоді задача ідентифікації полягатиме у пошуку такого $Ind \in \widetilde{P}^k, k = 1,\cdots,4$,
де $\widetilde{P}^k = \left\{\left(\widetilde{y}_i^k, Ind\right) : \left|y_j^k - \hat{y}^k\right| \le R_k,\, j = 1\cdots L,\, y_j^k \in P^k\right\}$,
де $\hat{P} = \left\{\hat{y}^k, k = 1,\cdots,4\right\}$ – множина ознак вхідного контуру для пошуку.

Радіус пошуку R підбирається із врахуванням деяких незбігів між вхідним контуром та контуром, що записано раніше.

Алгоритмічна реалізація підходу

Основна ідея підходу полягає у відображенні множини (12) у координатний простір [7]. Як обчислювальну платформу пропонується використати SQL-сервер. Використання традиційної реляційної моделі з властивою їй декомпозицією досить громіздке. Беручи до уваги, що множини (12) досить сталі, для проектування бази даних пропонується структура предметної області у вигляді *n*-вимірного куба [7].



Рисунок 1 – 2-вимірна структура

Matrix : таблица

A	B	Data1	Data2
0	1	Прізвище	
0	2	Ім'я	
0	3	По батькові	
0	4	Дата народження	
0	11	Точка-ознака 1 (у-координата)	
0	21	Точка-ознака 2 (у-координата)	
0	31	Точка-ознака 3 (у-координата)	
0	41	Точка-ознака 4 (у-координата)	
0	12	Максимальна відстань між точками у множині 1	
0	22	Максимальна відстань між точками у множині 2	
0	32	Максимальна відстань між точками у множині 3	
0	42	Максимальна відстань між точками у множині 4	

Рисунок 2 – Довідники

Предметна область задачі має єдиний об'єкт «людину» з рядом параметрів. Прізвище, ім'я, дата народження, координати точок-ознак – параметри об'єкта «людина». Всі параметри унікальні, тому не потрібно реалізовувати підлеглих структур. Для одного об'єкта з декількома параметрами *n*-вимірна структура вироджується у 2-вимірну з вимірами *A* та *B* (рис. 1). Розглянемо більш детально виміри. При *A = 0* значення даних на осі *B* є, по суті, довідниками (рис. 2). Значення виміру *A* від *1* до *L* (де *L* – кількість людей) призначені для фіксації інформації по конкретних людях (рис. 3). Для збереження, модифікації та пошуку даних (d) використовуватимемо:

$$d_i = (a_i, b_i), \tag{13}$$

де $a_i \in \{0,1..., L\}$, $b_i \in \{1, 2, 3, 4, 11, 21, 31, 41\}$, d_i – значення, які знаходяться за координатами (a_i, b_i).

Matrix : таблица

	A	B	Data1	Data2
▶	1	1	Іванов	
	2	1	Петров	
	3	1	Сидоров	

Рисунок 3 – Вимір А – інформація по людях

Matrix : таблица

	A	B	Data1	Data2
	1	1	Іванов	
	1	2	Іван	
	1	3	Іванович	
	1	4	01.03.1963	
	1	11		0.005
	1	21		0.491
	1	31		0.05
	1	41		-0.03

Рисунок 4 – Інформація по одній людині

Можливі комбінації a_i і b_i означатимуть: $(Ind,1)$ – прізвище, $(Ind,2)$ – ім'я, $(Ind,3)$ – по батькові, $(Ind, 4)$ – дата народження, $(Ind, 11)$ – y-значення першої точки-ознаки, $(Ind, 21)$ – y-значення другої точки-ознаки, $(Ind, 31)$ – y-значення третьої точки-ознаки, $(Ind, 41)$ – y-значення четвертої точки-ознаки ($Ind \in \{1,..., L\}$, де L – кількість людей. Беручи до уваги, що запропонована структура реалізовуватиметься за допомогою реляційної СУБД, і пошук виконуватиметься засобами цієї ж СУБД, то для оптимізації варто область значень подати двома стовпчиками: *Data1* – для нечислової інформації (текст, дата) та *Data2* – для числової інформації. Тоді стовпчик *Data2* можливо проіндексувати, що прискорить пошук, організований засобами реляційної СУБД. Приклад заповнення інформації по одній людині наведений на рис. 4.

Відображення потрібної нам предметної області потребує лише (на відміну від реляційної моделі) однієї таблиці. Засоби роботи з цією таблицею (stored-процедури з SQL-запитами) також суттєво спрощуються. Запис та модифікацію довільного рядка таблиці реалізує наступна stored-процедура (лістинг 1):

Лістинг 1 – Stored-процедура для запису та модифікації рядка у таблиці

```
CREATE PROCEDURE dbo.Insert
@A int,@B int,@Data1 Varchar(50),@Data2 float
AS
IF (SELECT COUNT(*) FROM Matrix WHERE (A=@A)AND(B=@B))>0
 BEGIN
  UPDATE Matrix SET A=@A,B=@B,Data1=@Data1,Data2=@Data2 WHERE (A=@A)AND(B=@B)
 END
ELSE
 BEGIN
  INSERT INTO Matrix (A,B,Data1,Data2) VALUES (@A,@B,@Data1,@Data2)
 END
GO
```

Визначення особи за чотирма вхідними точками-ознаками профілю потрібує додатково розрахунку максимальної відстані (радіусу пошуку) між двома сусідніми точками для кожної з чотирьох множин. Розрахунок реалізовано за наступним алгоритмом:

– *вхідна інформація*: значення *y*-координати нової точки *i*-ї множини y_n ($i = 1,\cdots,4$);

– шукаємо першу точку (y_1) за умовою $y_i > y_n$ та розраховуємо відстань: $d_1 = |y_1 - y_n|$;

– шукаємо другу точку (y_2) за умовою $y_i < y_n$ та розраховуємо відстань $d_2 = |y_2 - y_n|$;

– вибираємо максимальне значення відстані $d_{\max} = \max(d_1, d_2)$;

– якщо знайдене значення максимальної відстані більше за записане раніше (для відповідної множини), то модифікуємо його.

Наведений алгоритм реалізує відповідна stored-процедура (лістинг 2). Вона виконується після кожного запису або модифікації значень точок-ознак. Її результатом є модифікація рядків з координатами *(0,12)*, *(0,22)*, *(0,32)*, *(0,42)*, поле *Data2* яких містить поточне значення максимальної відстані для кожної множини.

Лістинг 2 – Stored-процедура для визначення максимальної відстані
між значеннями у відповідній множині (*@B*) точок-ознак

```
CREATE PROCEDURE dbo.ModificationDistance
@B int,@Yn float, @Data2 float
AS
IF (SELECT COUNT(*) FROM Matrix WHERE (A>0)AND(B=@B))>1
 BEGIN
  DECLARE @D float, @Max float
  SELECT @D=(SELECT MAX(y) FROM (
    SELECT TOP 1 ABS(Data2-@Yn) y FROM Matrix Where A=0 AND B=@B AND Data2>@Yn
    UNION
    SELECT TOP 1 ABS(Data2-@Yn) y FROM Matrix Where A=0 AND B=@B AND Data2<@Yn))
SELECT @MAX=(SELECT DATA2 FROM Matrix WHERE B=@B AND A=0)
  IF @Max<@D
   BEGIN
    UPDATE Matrix SET Data2=@D FROM Matrix WHERE (B=@B)AND(A=0))
   END
 END
GO
```

Ідентифікація особи за точками-ознаками профілю носа виконується за наступним алгоритмом:

– *вхідна інформація*: чотири *y*-координати точок-ознак шуканого контуру;

– для кожного значення вхідної *y*-координати у відповідній множині шукаються значення, що попадають в окіл (+/– максимальна відстань між двома значеннями у множині);

– шукається перетин отриманих множин за умови наявності у них значень однієї особи;

– якщо такий перетин не пустий – шукане знайдене, в іншому випадку – шукане не знайдене.

Наведений у лістингу 3 код stored-процедури реалізує ідентифікацію особи за описаним алгоритмом.

Лістинг 3 – Stored-процедура для ідентифікації особи

```
CREATE PROCEDURE dbo.Identification
-- вхідні значення
@Y1 float, @Y2 float, @Y3 float, @Y4 float,
-- вихідне значення
@Ind int
AS
-- визначаємо максимальні відстанні між значеннями для 4 множин y-координат точок-ознак
DECLARE @Dist1 float, @Dist2 float, Dist3 float, @Dist4 float
SELECT @Dist1=(SELECT Data2 FROM Matrix WHERE A=0 AND B=12)
SELECT @Dist2=(SELECT Data2 FROM Matrix WHERE A=0 AND B=22)
SELECT @Dist3=(SELECT Data2 FROM Matrix WHERE A=0 AND B=32)
SELECT @Dist4=(SELECT Data2 FROM Matrix WHERE A=0 AND B=42)
--визначаємо 4 підмножини з кандидатами на ідентифікацію
drop table #T1
CREATE TABLE #T1 ([A] [int] NOT NULL)
INSERT INTO #T1 SELECT A FROM Matrix Where ABS(Data2-Y1)<=Dist1 AND B=11
drop table #T2
CREATE TABLE #T2 ([A] [int] NOT NULL)
INSERT INTO #T2 SELECT A FROM Matrix Where ABS(Data2-Y2)<=Dist2 AND B=21
drop table #T3
CREATE TABLE #T3 ([A] [int] NOT NULL)
INSERT INTO #T3 SELECT A FROM Matrix Where ABS(Data2-Y3)<=Dist3 AND B=31
drop table #T4
CREATE TABLE #T4 ([A] [int] NOT NULL)
INSERT INTO #T4 SELECT A FROM Matrix Where ABS(Data2-Y4)<=Dist4 AND B=41
-- визначаємо результат ідентифікації як перетин
SELECT @Ind=(SELECT #T1.A as Ind FROM #T1,#T2,#T3,#T4 WHERE Ind=#T2.A AND Ind=#T3.A AND
Ind=#T4.A)
GO
```

Експеримент

Для перевірки спроможності запропонованого підходу була побудована множина із 17 контурів профілів носів (рис. 5).

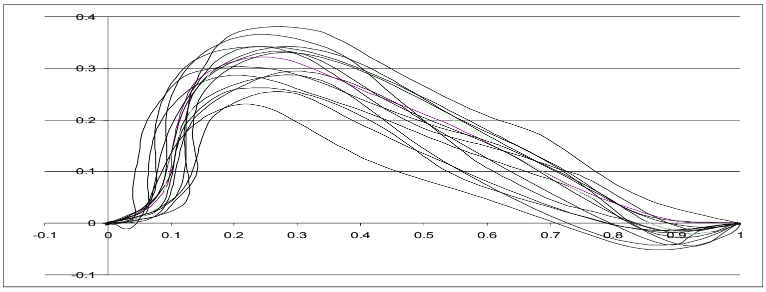


Рисунок 5 – Контури профілів носів

Для побудови цієї множини фотографічне зображення профілю людини переводилось до контурного вигляду, скелетизувалось, отримувались точки контуру носа та за допомогою перетворень (5) і (6) отримувались контури носів для подальшого аналізу (рис. 6).

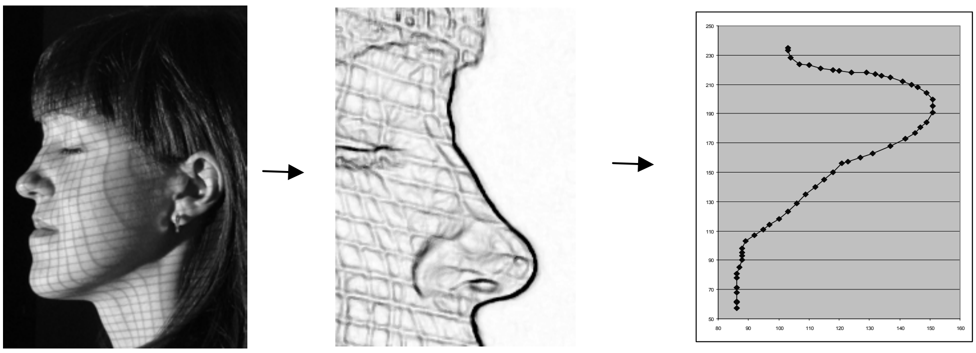


Рисунок 6 – Отримання контуру носа

Застосувавши перетворення (7) для цих контурів носів, було отримано чотири множини (12). Рис. 7 містить графічне відображення цих множин.

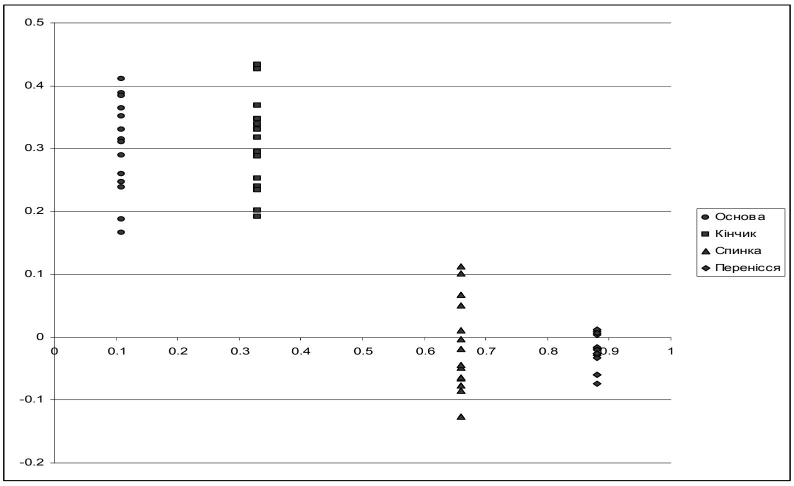


Рисунок 7 – Множини параметрів-особливостей для 17 носів

Висновки

Запропонований підхід показав свою ефективність для вирішення поставленої задачі. Подальші дослідження будуть направлені на вдосконалення підходу: тестування на більшій множині, яка б була репрезентативною для даної території; розгляд контурів носів близьких родичів тощо.

Література

1. Moorhouse A. The Nose on Your Face May Not be so Plain: Using the Nose as a Biometric / A. Moorhouse, A.N. Evans, G.A. Atkinson [and other] // IET 3rd International Conference on Imaging for Crime Detection and Prevention (ICDP-09), Kingston, UK, December 2009. – Режим доступу : http:// people.bath.ac.uk/ eesane/Nose%20Biometrics/Nose%20Biometrics%20Presentation.pdf
2. Леонардо да Винчи. О том, как изображать лицо, фигуру и одежды [Электронный ресурс] / Леонардо да Винчи. – Режим доступу : http://ofap.ulstu.ru/history/leonardo/doc/leonardo/leon_010.php.htm
3. Криминалистическое описание внешности человека (функциональные и сопутствующие элементы и признаки) : справочное пособие / [Зинин А.М., Виниченко И.Ф., Житников В.С., Овсянникова М.Н.] ; под ред. проф. В.А. Снеткова. – М. : ВНИИ МВД СССР, 1988. – 242 с.
4. Piegl L. The NURBS Book / L. Piegl, W. Tiller. – 2nd Edition. – Berlin, Germany : Springer-Verlag, 1996. – 646 p.
5. Лоусон Ч. Численное решение задач метода наименьших квадратов / Ч. Лоусон, Р. Хенсон. – М. : Наука, 1986. – 232 с.
6. Роджерс Д. Математические основы машинной графики / Д. Роджерс, Дж. Адамс. – М. : Мир, 2001. – 604 с.
7. Бармак О.В. Подход для моделирования структуры предметной области в виде n-мерного куба / О.В. Бармак, М.Л. Яновський // УСИМ. – 2010. – № 5.

А.В. Бармак, Ю.В. Крак, Ю.В. Кривонос

Об одном подходе к идентификации личности по контуру профиля носа

В статье рассматривается подход для идентификации личности по контуру носа. Для нахождения пространства признаков предлагается использовать аппроксимацию с помощью *B*-сплайнов. Алгоритмическая реализация подхода использует моделирование структуры базы данных в виде *n*-мерного куба.

O.V. Barmak, Iu.V. Krak, Iu.G. Kryvonos

About one Approach of Human Identification by the Nose Contour Profile

An approach of human identification by the nose contour profile is researched in the paper. *B*-splines approximation is suggested for determining the feature space. Algorithmic implementation of the approach uses database structure in the form of *n*-dimensional cube.

Стаття надійшла до редакції 19.07.2010.